Amendments to the Claims

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of claims:

1 – 11. (cancelled)

12. (currently amended) A method for verifying a signature for a message *m* in a data communication system established between a sender and a ~~recipient~~ verifier, said sender ~~generating~~ having generated in a secure computer system a masked signature ~~components (r, s, e), where~~ having a first signature component r ~~is an integer derived from a coordinate of~~ computed using a first short term public key [[kP,]] derived from a first short term private key; a second signature component s ~~is a signature component derived by binding~~ computed using a second short term private key[[, the]] on said message *m*, ~~and short and long term private keys, and~~ a long term private key, and said first signature component r; and a third signature component c ~~is a second signature component obtained by combining~~ computed using said first and second short term private keys, said method for verifying comprising ~~the steps of a~~ said verifier:

a) obtaining ~~a pair of signature components $(\bar{s},r)$, said component $\bar{s}$ being derived from said first and second signature components generated by a signer~~ a regular signature derived from said masked signature (r, s, c), said regular signature having said first signature component r, and another signature component $\bar{s}$ computed using said second signature component s and said third signature component c;

b) recovering a ~~coordinate pair (x_t,y_t) corresponding to said first short term public key kP using said pair $(\bar{s},r)$ and~~ point on an elliptic curve defined over a finite field using said message *m* and said another signature component $\bar{s}$;

c) converting an element of said point to an integer;

[[c)]]d) calculating a ~~signature component~~ value *r'* ~~from one of said coordinate pair~~ using said integer; and

[[d)]]e) verifying said regular signature $(r,\bar{s})$ if said value r' [[=]] is equal to said first signature component r.

13. (currently amended) A method according to claim 12 further comprising the step of said verifier receiving said masked signature (r, s, c) from said ~~signor~~ sender and converting ~~(s, r, c)~~ (r, s, c) to obtain said ~~pair $(\bar{s},r)$~~ regular signature $(r,\bar{s})$.

14. (currently amended) A method according to claim 12 further comprising the step of said ~~signor~~ sender converting said masked signature ~~(s, r, c)~~ (r, s, c) to said ~~pair $(\bar{s},r)$~~ regular signature $(r,\bar{s})$ and said ~~signor~~ sender sending said ~~pair $(\bar{s},r)$~~ regular signature $(r,\bar{s})$ to said verifier.

15. (currently amended) A method according to claim 12 wherein said ~~coordinate pair (x_1,y_1)~~ point is calculated using a pair of values u and v, said values u and v derived from said ~~pair $(\bar{s},r)$~~ regular signature $(r,\bar{s})$ and said message m.

16. (currently amended) A method according to claim 15 wherein said ~~coordinate pair (x_1,y_1)~~ point is calculated as $(x_1,y_1) = uP + vQ$, wherein P is a point on an elliptic curve E and Q is a public verification key of said ~~signor~~ sender derived from P as $Q = dP$.

17. (previously presented) A method according to claim 15 wherein said value u is computed as $u = \bar{s}^{-1}e \bmod n$ and said value v is computed as $v = \bar{s}^{-1}r \bmod n$, e being a representation of said message m.

18. (currently amended) A method according to claim 17 wherein e is calculated as $e = H(m)$, H( ) being a hash function of said ~~signor~~ sender and being known to said verifier.

19. (currently amended) A method according to claim 12 wherein [[said]] a coordinate x_1 of said point is first converted to an integer $\bar{x}_1$ prior to calculating said component r'.

20. (previously presented) A method according to claim 19 wherein said component r' is calculated as $r' = \bar{x}_1 \bmod n$.

21. (currently amended) A method according to claim 12 wherein prior to calculating said component r', [[said]] a coordinate pair (x_1,y_1) of said point is first verified, whereby if said coordinate pair (x_1,y_1) is a point at infinity, then said regular signature $(r,\bar{s})$ is rejected.